

w.c.s. - Development #12457

bibliothèque de calcul/code

06 juillet 2016 09:59 - Frédéric Péters

Statut:	Fermé	Début:	06 juillet 2016
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:	v1.56	Planning:	
Patch proposed:	Oui		
Description			
Comme les data sources et les webservices, titre, slug, code. (le code idéalement placé dans un textarea en monospace qui gère les tabulations en insérant quatre espaces) Ajouter de quoi appeler ça via ezt (genre [calc slug param1 param2 param3]).			

Révisions associées

Révision 8c6bd68a - 19 août 2016 14:30 - Frédéric Péters

general: add support for custom scripts in context variables (#12457)

Historique

#1 - 02 août 2016 12:09 - Frédéric Péters

Finalement ça m'ennuierait quand même vraiment d'ajouter cet espace de code éditable à travers le web; ce que je proposerais plutôt c'est que dans un sous-répertoire du tenant on puisse poser des scripts.

#2 - 12 août 2016 12:26 - Frédéric Péters

- Fichier 0001-general-add-support-for-custom-scripts-in-context-va.patch ajouté
- Statut changé de Nouveau à En cours
- Patch proposed changé de Non à Oui

#3 - 19 août 2016 11:55 - Thomas Noël

Ok avec ça, modulo un `s[script.hello earth]/[script.hello "earth"]/` dans la doc.

Peut-être ajouter aussi une docstring dans l'exemple car je pense que rapidement il faudra une `/backoffice/scripts` qui liste les scripts disponibles dans l'instance (avec, donc, affichage de leur docstring)

#4 - 19 août 2016 12:29 - Frédéric Péters

Je viens de passer du temps sur le côté docstring et ça va être galère dans la forme actuelle; je vais reprendre le code pour fonctionner différemment : utiliser `imp` pour charger le fichier comme si c'était un module.

```
imp.load_module("", open('/path/to/filename.py'), 'filename.py', ("", imp.PY_SOURCE))
```

Et demander que le code ait une fonction "main". Ça permettra alors d'atteindre la fonction `module.main` et sa docstring `module.main.__doc__`.

#5 - 19 août 2016 12:44 - Frédéric Péters

En fait non, ça sera simple, il suffira à l'endroit où on voudra afficher la documentation de ne pas passer par le `code_object` mais de faire un `ast.get_docstring(ast.parse(...))`.

#6 - 19 août 2016 12:54 - Frédéric Péters

- Fichier 0001-general-add-support-for-custom-scripts-in-context-va.patch ajouté

#7 - 19 août 2016 14:15 - Frédéric Péters

- Fichier 0001-general-add-support-for-custom-scripts-in-context-va.patch ajouté

Avec une correction d'orthographe à la documentation.

#8 - 19 août 2016 14:23 - Thomas Noël

Ack

(et je me dis qu'on affichera dans /backoffice/script uniquement les scripts qui ont une docstring, histoire de)

#9 - 19 août 2016 14:37 - Frédéric Péters

- Statut changé de *En cours* à *Résolu* (à déployer)

```
commit 8c6bd68a76c05acbe9b011584945b62dlb2e2ae3
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Fri Aug 12 12:01:56 2016 +0200
```

```
general: add support for custom scripts in context variables (#12457)
```

#10 - 04 décembre 2016 12:55 - Frédéric Péters

- Version cible mis à *v1.56*

#11 - 04 décembre 2016 13:12 - Frédéric Péters

- Statut changé de *Résolu* (à déployer) à *Fermé*

Fichiers

0001-general-add-support-for-custom-scripts-in-context-va.patch	9,66 ko	12 août 2016	Frédéric Péters
0001-general-add-support-for-custom-scripts-in-context-va.patch	9,98 ko	19 août 2016	Frédéric Péters
0001-general-add-support-for-custom-scripts-in-context-va.patch	9,98 ko	19 août 2016	Frédéric Péters